



## Monte Carlo superposition calculations of resonance integrals in a reactor cell

Kirkegaard, P.

*Publication date:*  
1970

*Document Version*  
Publisher's PDF, also known as Version of record

[Link back to DTU Orbit](#)

*Citation (APA):*  
Kirkegaard, P. (1970). *Monte Carlo superposition calculations of resonance integrals in a reactor cell*. Risø National Laboratory. Risø-M No. 1257

---

### General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Fi 25-304

C

We regret that some of the pages in the microfiche copy of this report may not be up to the proper legibility standards, even though the best possible copy was used for preparing the master fiche.

## CONTENTS

	Page
1. Introduction .....	3
2. Statement of the Problem .....	3
3. The Resonance Cross Sections .....	4
4. The Superposition Principle .....	6
5. Monte Carlo Sampling in MCSUP .....	13
5.1. Sampling of Initial Energy .....	13
5.2. Surface Source Sampling .....	14
5.3. Coveyou Sampling from Truncated Exponential .....	14
5.4. Conventional Transport Sampling .....	15
5.5. Choice of Estimator .....	15
5.6. Statistical Processing .....	18
6. Calculation with the MCSUP Program .....	18
6.1. Preparation of Input Data .....	18
6.2. Interpretation of Output .....	17
6.3. Solution of Various Problems by MCSUP .....	18
7. Possible Generalisations .....	20
8. Summary and Conclusions .....	20
9. Acknowledgement .....	21
10. References .....	21
11. Appendix I: MCSUP Data Sheet .....	24
12. Appendix II: Result for Sample Problem .....	25
13. Appendix III: Computer Code Print-out .....	28
14. Appendix IV: Escape Probability $P_e$ for Infinite Cylinder .....	41
15. Appendix V: Fuel-to-Moderator Collision Probability $P^*$ .....	42
16. Appendix VI: A Reluctive Integration Routine .....	47
17. Appendix VII: A Fast Exponential Routine .....	47
18. Appendix VIII: Optimisation of Computer Time for the Four Types of Starters .....	49

$$p = \exp$$

$$\langle t_0 \rangle$$

Page

19. Appendix IX: Computer Dependent Features of MCSUP .....	51
19.1. The Timing Routine .....	51
19.2. The Random Number Generator .....	51
19.3. Input/Output Units .....	52

Figures

## 1. INTRODUCTION

In a forthcoming report, Mikkelsen<sup>1)</sup> describes a computer program RESAB which carries out a multigroup calculation of the resonance absorption and flux spectrum in a reactor cell, using collision probability technique. His calculation scheme relies on the validity of certain approximations making possible very efficient computing. In developing such an approximate model there arises a natural need of checking the model or parts of it against some exact calculation method; in view of the complexity of the problem Monte Carlo seems to be the only practical way of performing such reference calculations.

This paper describes a Monte Carlo FORTRAN IV computer program, AEK P-598 MCSUP, designed to calculate resonance integrals by the method of superposition. The superposition principle was developed and first put into practical operation by Spanier and Gelbard<sup>2,3)</sup>. MCSUP works fairly efficiently for most problems, and it is possible that it may be valuable also for other purposes than comparative.

## 2. STATEMENT OF THE PROBLEM

Consider an infinite-cylindrical reactor lattice cell with a cross-section as shown in fig. 1. The heterogeneous resonance capture problem is idealized to a two-zone configuration, the cladding zone being neglected. The fuel cylinder radius is  $a$ , and the lattice pitch is denoted  $d$ . The cross sectional areas of the two zones are  $A_1 = \pi a^2$  and  $A_2 = \pi d^2 - A_1$ .

The fuel zone contains only one isotope (e. g. U-235), while the moderator may be composed of several nuclide types. Scattering in the fuel is assumed to proceed isotropically in the L-system, scattering in the moderator isotropically in the L-system or in the CM-system depending on the user's choice. No absorption is allowed in the moderator. The cell is assumed to be a part of an infinite lattice, so that the boundary conditions are periodic. The problem is to calculate the resonance integrals and the capture probabilities for one or more nuclides. The calculation is performed by the Monte Carlo method. The results are compared to the results of the deterministic method.



#### 4. THE SUPERPOSITION PRINCIPLE

In this section we shall mainly follow Spanier and Gelbard<sup>2)</sup> in giving a brief exposition of the superposition principle when applied to resonance absorption calculations.

The superposition principle states that the flux produced by the sum of two sources is the sum of the fluxes produced, separately, by each source. This follows immediately from the linearity of the transport equation. The idea is now that the problem to be solved ("problem  $\alpha$ ") is superimposed by another problem ("problem  $\beta$ ") in such a way that the sum problem has a simple analytical solution. Hence, the Monte Carlo work is confined to problem  $\beta$ , and sometimes this method of attack leads to significant reduction of variance.

This basic approach has to be extended a little in order to yield an efficient model for resonance calculations. Consider again fig. 1 showing the fuel zone  $R_I$  and the moderator zone  $R_{II}$  with a reflecting boundary. As before we wish to solve problem  $\alpha$ . The source  $Q_\alpha(E, r)$  in this problem is a slowing-in density in both zones from above  $E_u$ . Problem  $\alpha$  has the solution (angular flux)  $F_\alpha = F_\alpha(E, r, \omega)$ , where  $E$  is energy,  $r$  position, and  $\omega$  the unit vector of direction. Above  $E_u$ ,  $F_\alpha$  is assumed to be the unperturbed  $1/E$ -flux. We normalize in such a way that  $F_\alpha = \frac{1}{4\pi E}$  for  $E > E_u$ .

In the following it is essential to distinguish between resonance and potential scattering in the fuel. The macroscopic fuel cross section is therefore considered as composed by one absorption and two scattering terms:

$$\Sigma_{Tf}(E) = \Sigma_{af}(E) + \Sigma_{sf}(E) = \Sigma_{af}(E) + \Sigma_{srf}(E) + \Sigma_{pf}(E) = \Sigma_{srf}(E) + \Sigma_{pf}(E) \quad (11)$$

We now define a problem  $\beta$  with the same geometry but with a source (uniform or localized) in the moderator. The spectral cross sections are calculated in the interval  $E_1 < E < E_u$  before the Monte Carlo work in the fuel. No source is present in the moderator. Let the problem  $\beta$

have the solution  $F_\beta$ .

The sum problem of  $\alpha$  and  $\beta$  has the solution  $F_T$ . Now, it is not  $F_T$  itself that has a simple expression but that part of it,  $F_{Tu}$ , which has not yet undergone resonance scatterings. In fact,

$$F_{Tu}(E, r, \omega) = \frac{1}{4\pi E}, \quad E_L < E < E_u \quad (13)$$

To see this we write down the transport equations satisfied by  $F_{Tu}$  in  $R_I$  and  $R_{II}$ :

$$L F_{Tu}(E, r, \omega) = \int d\omega' \int_{E_L}^{E_u} dE' \Sigma_{Pf}(\omega, \omega', E' - E, r) F_{Tu}(E', r, \omega') + \frac{1}{4\pi} Q_\beta(E, r) + \frac{1}{4\pi} Q_\beta(E), \quad r \in R_I$$

and

$$L F_{Tu}(E, r, \omega) = \int d\omega' \int_{E_L}^{E_u} dE' \Sigma_S(\omega, \omega', E' - E, r) F_{Tu}(E', r, \omega') + \frac{1}{4\pi} Q_\beta(E, r), \quad r \in R_{II} \quad (14)$$

$\Sigma_{Pf}(\omega, \omega', E' - E, r)$  is the differential cross section for potential scattering in the fuel. The operator  $L$  is defined by

$$L = \omega \cdot \nabla + \Sigma_{af}(E) F_{Tu}(E, r, \omega) = 0 \quad (15)$$

Otherwise the notation is conventional.

In (14) all scattering cross sections are energy independent. The notation of (13) and (14) is conventional. The quantity is denoted by  $F_{Tu}$  in the interval  $E_1 < E < E_u$  before the Monte Carlo work in the fuel. No source is present in the moderator. Let the problem  $\beta$

$$F_a = F_{Tu} + F_{TS} - F_\beta = \frac{1}{4\pi E} + F_{TS} - F_\beta \quad (16)$$

It is now practical to split up  $F_\beta$  into four components, according to the history of the flux. Let

$$\begin{aligned} F_\beta &= F_\beta \text{ (unscattered before exit, unreturned)} \\ &+ F_\beta \text{ (unscattered before exit, returned)} \\ &+ F_\beta \text{ (resonance scattered before exit)} \\ &+ F_\beta \text{ (potential scattered before exit), or} \end{aligned}$$

$$F_\beta = F_{\beta uu} + F_{\beta ur} + F_{\beta sr} + F_{\beta sp} \quad (17)$$

"Exit" means here absorption in or escape from the fuel. By (16) and (17),

$$F_a = \frac{1}{4\pi E} - F_{\beta uu} - F_{\beta ur} - F_{\beta sp} + F_{TS} - F_{\beta sr} \quad (18)$$

We are interested in calculating the resonance capture probability  $1-p$ , or, equivalently, the mean absorption rate per ccm in the fuel

$$A_T = \frac{1}{A_1} \int_{E_L}^{E_u} dE \int_I dr \Sigma_{af}(E) \Phi_a(E, r) \quad (19)$$

where  $\Phi$ , of course, stands for a scalar flux and  $I$  symbolizes the two-dimensional fuel region  $R_1$ . The connection between  $A_T$  and  $1-p$  is

$$1-p = \frac{A_T}{A_1} \quad (20)$$

where  $A_1$  is the total absorption rate per ccm in the fuel

$$\langle \Sigma_a \rangle = \frac{1}{V} \int_V dV \Sigma_a(E, r) \quad (21)$$

and  $V$  is the volume of the fuel region.

$$\langle \Sigma_a \rangle = \frac{NA_1 \langle \Sigma_a \rangle}{A_1 + A_2} \quad (21)$$

where  $N$  is the number density of the absorber.

Defining

$$I'_0 = \frac{1}{A_1} \int_{E_L}^{E_u} dE \int_I dr \Sigma_{af}(E) \left[ \frac{1}{E} - \Phi_{\beta uu}(E, r) \right] \quad (22)$$

$$I'_1 = \frac{1}{A_1} \int_{E_L}^{E_u} dE \int_I dr \Sigma_{af}(E) \Phi_{\beta ur}(E, r) \quad (23)$$

$$I_2 = \frac{1}{A_1} \int_{E_L}^{E_u} dE \int_I dr \Sigma_{af}(E) \Phi_{\beta sr}(E, r) \quad (24)$$

$$J = \frac{1}{A_1} \int_{E_L}^{E_u} dE \int_I dr \Sigma_{af}(E) \left[ \Phi_{TS}(E, r) - \Phi_{\beta sr} \right] \quad (25)$$

then

$$A_T = I'_0 - I'_1 - I_2 + J \quad (26)$$

It can be shown that

$$I'_0 = \int_{E_L}^{E_u} dE \Sigma_{af}(E) \left[ P_0(E) + \frac{P_1}{E} (1-P_0(E)) \right] \quad (27)$$

where  $P_0$  is the collision probability from fuel to moderator.

where  $P_1$  is the collision probability from moderator to fuel.

where  $P_2$  is the collision probability from fuel to fuel.

where  $P_3$  is the collision probability from moderator to moderator.

where  $P_4$  is the collision probability from fuel to moderator.

where  $P_5$  is the collision probability from moderator to fuel.

where  $P_6$  is the collision probability from fuel to fuel.

where  $P_7$  is the collision probability from moderator to moderator.

where  $P_8$  is the collision probability from fuel to moderator.

where  $P_9$  is the collision probability from moderator to fuel.

It can be shown that  $I'_1$  is the absorption rate produced by an outgoing source on the surface of the fuel

$$Q_{1P}(E, \omega) = \frac{\Sigma_{rf}(E)}{4\pi E} \frac{\mu(1 - \exp(-l \cdot \Sigma_{Tf}(E)))}{\Sigma_{Tf}(E)} \quad (28)$$

Here,  $\mu = \omega \cdot n > 0$ ,  $n$  being the outward unit normal at the surface point  $P$ , and  $l$  is the backward intersection path from  $P$  in the fuel (due to the scattering interference,  $Q_{1P}(E, \omega)$  may become negative at some energies, this tending to increase variance somewhat). Integrating (28) over directions gives  $Q'_1(E) = \int Q_{1P}(E, \omega) d\omega$ , which is proportional to the density function for selecting starting energies. It follows that <sup>2)</sup>

$$Q'_1(E) = \frac{\Sigma_{rf}(E)}{E} \cdot P_0(E) \quad (29)$$

The Monte Carlo calculation of  $I'_1$  includes such starters which have been absorbed on their first collision. This contribution will in our model be deleted from  $I'_1$  and included in the deterministic term  $I'_0$ . In other words, we write

$$I'_0 - I'_1 = I_0 - I_1 \quad (30)$$

where now  $I_0$  is to be evaluated deterministically and  $I_1$  by Monte Carlo. One can show that <sup>2)</sup>

$$I_0 = \int_{E_L}^{E_H} \frac{dE}{E} \Sigma_{af}(E) \left[ P^*(E) + \frac{\Sigma_{Pf}}{\Sigma_{Tf}(E)} (1 - P^*(E)) \right] \quad (31)$$

where  $P^*$  is the collision probability <sup>2, 6)</sup> from fuel to moderator.  $P^*$  is discussed in Appendix V, and the quadrature evaluation of  $I_0$  in Appendix VI. Going from  $I'_1$  to  $I_1$  (29) is changed to  $I_1 = I'_1 - I'_0 + I_0$ . The Monte Carlo approach is confined to the computation of  $I_1$ .

$$Q_1(E) = \frac{\Sigma_{rf}(E)}{E} \left[ P_0(E) - \frac{\Sigma_{af}(E)}{\Sigma_{Tf}(E)} (P_0(E) - P^*(E)) \right] \quad (32)$$

Hence, to sample  $I_1$  we have to pick the starting energy from (32) and then sample an  $\omega$  from (28). How these steps are carried out in detail is discussed in sec. 5. However, if a starter is absorbed on its first collision, it is rejected; a new  $\omega$  is sampled from (28) without changing  $E$ .

The next term in (26),  $I_2$ , is generated by neutrons which are potential scattered before exit. The mean rate of such scatterings are

$$Q_2(E) = \frac{\Sigma_{rf}(E)}{E} (1 - P_0(E)) \frac{\Sigma_{Pf}}{\Sigma_{Tf}(E)} \quad (33)$$

From  $|Q_2(E)|$  we pick a starting energy. Then a starting point  $P$  is selected uniformly in the fuel and an  $\omega$  with isotropic distribution. If the starter leaves the fuel on its first flight it is rejected; without changing  $E$  we sample a new  $P$  and a new  $\omega$ . Otherwise we regard its first collision as a potential scattering. Subsequent sampling proceed conventionally.

The last term to be computed by Monte Carlo is  $J$  (eq. (25)). We show that this term requires two Monte Carlos. Let us define

$$F_{AS} = F_{TS} - F_{\beta sr} \quad (34)$$

By writing down the transport equations for  $F_{TS}$  and  $F_{\beta sr}$  one sees that  $F_{AS}$  satisfies the equations

$L F_{AS}(E, \mathbf{r}, \omega) = \int d\omega' \int d\mathbf{r}' \Sigma_{Tf}(\mathbf{r}, \omega, \omega') F_{AS}(E, \mathbf{r}', \omega')$   
 $F_{AS}(E, \mathbf{r}, \omega) = 0$  if  $\mathbf{r}$  is on the surface of the fuel.  
 The first history uses a sample value  $\xi_1$  for  $\xi$ . The second history uses  $\xi_2 = 1 - \xi_1$ . The third history uses a new sample value  $\xi_3$ , then  $\xi_4 = 1 - \xi_3$ , and so on. This method of antithetic variates <sup>2, 8)</sup> may be applied here in a very simple way to obtain variance reduction: the first history uses a sample value  $\xi_1$  for  $\xi$ , the second history uses  $\xi_2 = 1 - \xi_1$ , the third history uses a new sample value  $\xi_3$ , then  $\xi_4 = 1 - \xi_3$ , and so on. This method of antithetic variates <sup>2, 8)</sup> may be applied here in a very simple way to obtain variance reduction: the first history uses a sample value  $\xi_1$  for  $\xi$ , the second history uses  $\xi_2 = 1 - \xi_1$ , the third history uses a new sample value  $\xi_3$ , then  $\xi_4 = 1 - \xi_3$ , and so on.



$$L F_{AS}(E, r, \omega) = \int d\omega' \int_{E_L}^{E_U} dE' \Sigma_S(\omega, \omega', E' \rightarrow E, r) F_{AS}(E', r, \omega'),$$

$$r \in R_{II} \quad (35)$$

Hence,  $F_{AS}$  is generated by resonance scattering in  $R_I$  in the flux  $F_{Tu} - F_{Su}$ . One can show that this latter quantity itself satisfies a transport equation,

$$L [F_{Tu}(E, r, \omega) - F_{Su}(E, r, \omega)] = \frac{\Sigma_{PI}}{4\pi E}, \quad r \in R_I \quad (36)$$

and

$$F_{Tu}(E, r_P, \omega) - F_{Su}(E, r_P, \omega) = \frac{1}{4\pi E} \quad (37)$$

where  $r_P$  is a point of the fuel-moderator interface, and  $\omega$  is an inward direction ( $\omega \cdot n < 0$ ).

Eqs. (36) and (37) reflect the fact that  $F_{Tu} - F_{Su}$  is produced by the joint action of a uniform isotropic volume source  $\Sigma_{PI}/4\pi E$  and an inward-directed surface source with strength  $\omega \cdot n/4\pi E$ .

This leads to the following procedure for the calculation of the corresponding components,  $I_3$  and  $I_4$ , of  $J$ . In the  $I_3$ -case we write

$$Q_3(E) = \frac{\Sigma_{PI}}{E} (1 - P_0(E)) \frac{F_{Su}(E)}{F_{Tu}(E)}, \quad (38)$$

select an energy from  $|Q_3(E)|$  and proceed just as in the  $I_2$ -case (cf. eq. (33) et seq.), but here we force the starter to resonance-scatter in the fuel.

In the  $I_4$ -case we write

$$Q_4(E) = \frac{F_{Su}(E)}{F_{Tu}(E)} \quad (39)$$

select an energy from  $|Q_4(E)|$  and a starting point from a distribution uniform over the surface. The starting direction has to be inward, but

otherwise this sampling is quite analogous to the  $I_1$ -case. The first collision is forced to be a resonance scattering in  $R_I$ . Hereafter, the sampling proceeds conventionally.

To recapitulate, we have expressed the rate of absorption  $A_f$  as a sum of five terms

$$A_f = I_0 - I_1 - I_2 + I_3 + I_4 \quad (40)$$

where  $I_0$  can be evaluated by quadrature, but  $I_1, I_2, I_3, I_4$  each requires one Monte Carlo. In the following section we discuss some problems in connection with the Monte Carlo sampling.

## 5. MONTE CARLO SAMPLING IN MCSUP

### 5.1. Sampling of Initial Energy

From the formulas (32), (33), (38), (39), it is seen that the Monte Carlo for each of the four terms  $I_1, I_2, I_3, I_4$ , requires an initial energy sampling from a density function proportional to some known function  $Q(E)$ . How this is done is shown schematically in fig. 2.  $Q(E)$  is approximated by a step function corresponding to a division of the energy interval  $E_L \leq E \leq E_U$  in 1000 equal subintervals (only 8 shown in fig. 2) with step heights equal to  $Q(E)$  taken in the subinterval midpoints. This matches the discrete cross section evaluation mentioned in sec. 3. The step function is normalized to a density function and is next converted to a distribution function  $F(E)$  of the polygon type. After selection of a random number  $\xi$ ,  $E$  is given by  $F^{-1}(\xi)$ .

To increase speed we prefer not to use  $\xi$  itself, but the nearest discrete representation  $\xi_i$  corresponding to a division of the interval  $0 < \xi < 1$  in 1000 equal subintervals (only 8 shown in fig. 2). The values  $E = F^{-1}(\xi_i)$  are calculated and stored before the Monte Carlo is executed. Of course, this procedure involves a sacrifice in some precision.

As pointed out in ref. 2 the technique of arithmetic variables can be applied here in a very simple way to obtain variance reduction and save history space. Suppose we have calculated  $I_1$  for the calculation of  $A_f$ . The discrete energy values  $\xi_i$  are stored. The next Monte Carlo run is for the calculation of  $I_2$ . The discrete energy values  $\xi_j$  are stored. The next Monte Carlo run is for the calculation of  $I_3$ . The discrete energy values  $\xi_k$  are stored. The next Monte Carlo run is for the calculation of  $I_4$ . The discrete energy values  $\xi_l$  are stored. This technique was first introduced by [ref. 2].

## 5.2. Surface Source Sampling

Having selected the initial energy one turns to sample the source position and direction. Starters of types 2 and 3 (cf. sec. 4) have an isotropic and uniform volume distribution which is easily sampled, while the types 1 and 4 have an outgoing, resp. ingoing surface source distribution, which is considered in more detail now. As the surface positions are selected from a uniform distribution, we need only discuss the direction sampling. Because the sampling is quite similar in the two cases, we consider only the outgoing source (type 1).

The relevant sampling formula is (28), which now, when position and energy is known, reduces to

$$Q(\omega) = \mu [1 - \exp(-l(\omega) \cdot \Sigma)] \quad (41)$$

apart from a constant factor. As before,  $\mu = \omega \cdot n$ , and  $l(\omega)$  is the back-wards intersection path in the fuel.

A straightforward way would be to sample  $\omega$  from either an isotropic or a cosine distribution and adjust the weight of the starter according to (41). Fluctuating weights have, however, a deleterious influence on variance, unless the fluctuations reflect the importance of the events<sup>2)</sup>.

A better alternative seems here to be a cosine sampling combined with rejection<sup>2, 10)</sup>. As we then sample from a distribution proportional to  $\mu$ , the quantity in square brackets in (41) governs the rejection. Hence we pick a random number  $\zeta$  and accepts or rejects the starter according to the events  $1 - \exp(-l(\omega) \cdot \Sigma) \geq \zeta$ . In case of rejection a new  $\omega$ -sampling is carried out. The efficiency for this rejection is supposed to be high, because the rod for most initial energies is fairly black<sup>2)</sup>.

## 5.3. Coveyou Sampling from Truncated Exponential

As mentioned in sec. 4, starters of type 4 are forced to resonance-scatter in the fuel. This leads naturally to sampling from a truncated exponential with the density function

$$f(x) = \frac{e^{-x}}{1 - e^{-1}} \quad 0 \leq x \leq 1 \quad (42)$$

where  $x$  is the distance from the starting point to the next collision. The starting direction has to be inward, but

where 1 is the path from the source point to the rod intersection point along the flight. With a random number  $\zeta$ , the standard sampling formula corresponding to (42) becomes

$$x = -\frac{1}{\Sigma} \log [1 - \zeta(1 - \exp(-\Sigma))] \quad (43)$$

involving both a log- and an exp-calculation. Here, an alternative sampling formula due to Coveyou is used<sup>2, 16, 20)</sup>:

$$x = \frac{1}{\Sigma} \text{UDR} \left[ \frac{-\log \zeta}{\Sigma} \right] \quad (44)$$

UDR stands for the undivided remainder, e.g.  $\text{UDR} \left( \frac{13.83}{6} \right) = 1.83$ .

## 5.4. Conventional Transport Sampling

Apart from the special methods discussed above and the choice of estimator (see 5.5), the sampling of the neutron transport and energy degradation processes elapses quite conventionally and will not be discussed here; details will be found elsewhere<sup>2, 14, 16)</sup>. However, it should be quite easy to follow the various steps on the computer code print-out reproduced in Appendix III; it is hoped that the rather numerous comments will facilitate this work.

## 5.5. Choice of Estimator

Before the program reached its final form, some experimentation was done to find an efficient estimator for the resonance absorption. Three estimators were tried.

- 1) The Wasov estimator<sup>2)</sup> with forbidden absorption and corresponding weight reductions at collisions. History termination only by energy degradation.
  - 2) The analog estimator<sup>2)</sup> which is substantially independent of  $\Gamma$  for a given computer and is a measure of the efficiency of the estimator.
  - 3) The modified track length estimator.
- From our calculations it appeared that the modified track length estimator was the most efficient. This estimator was then introduced by the program.

track length estimator refers to an analog random walk process. Scoring takes place every time a neutron collides in, or escape from the fuel, after the following rule:

- When a neutron starts at the fuel point P (a source or scattering point, or a point on the fuel boundary) and thereafter escapes uncollided at E or scatters at S, then the scoring is  $\Sigma_a \cdot d$  where d is the track length PE or PS.
- When a neutron starts at the fuel point P and is absorbed in the fuel at A (whereby the history is terminated), then the scoring is

$$\frac{\Sigma_a}{\Sigma_t} \cdot \frac{1 - \exp(-\Sigma_t d) \{ \Sigma_t d + 1 \}}{1 - \exp(-\Sigma_t d)} \quad (45)$$

where  $d = PE$ , E denoting the hypothetical escape point if no collision had occurred. (45) is constructed by averaging the estimator  $\Sigma_a \cdot PA$  over all possible A<sup>2</sup>.

## 5.6. Statistical Processing

In order to carry out a statistical analysis of the Monte Carlo data, histories are collected in groups of 500 each. How many groups that are processed depends on the user's choice of computing time (see 6.1). Group averages are calculated, and from these a final average and a sample variance is found (a confidence analysis based on Student's distribution is not carried out). Finally, the "statistical quality" Q is calculated (see 6.2).

The statistical procedure has to be modified a little by the introduction of the optimization device discussed in Appendix VIII.

## 6. CALCULATION WITH THE MCSUP PROGRAM

### 6.1. Preparation of Input Data

Input data for the FORTRAN IV program P-598 MCSUP is prepared according to the input sheet given in Appendix I. This input sheet is for a single problem consisting of a fuel with

pure hydrogen as the moderator, and anisotropic moderator scattering is taken into account. Format and type of data items are derived from the sheet. The first item is the problem no; a negative problem no. stops the computations. The two integers in the next line are: random initial number (any nine-digit integer will do) and the number m of different nuclide types in the moderator (here  $m = 1$ ). The next two lines in the sheet contain real numbers; they are (in the notation previously used):  $E_0$ ,  $E_u$ ,  $E_L$ ,  $\sigma_p$ ,  $\Gamma_n$ ,  $\Gamma_v$ , A, T, a, d, number density of absorber  $\times 10^{24}$ , maximum time  $t_{max}$ . All energies are in eV.  $t_{max}$  is given in minutes and refers to the time spent in the Monte Carlo phase of the problem; to be safe, the user should not choose a  $t_{max}$  greater than  $t_{job} - 2$  where  $t_{job}$  is the maximum time in minutes stated on the job request card (this rule is relevant for calculations on the IBM 7094 at NEUCC in Denmark; modifications may be necessary for other computing centers). Hereafter come m data lines, one for each nuclide type in the moderator. Each line contains three real numbers and one integer: atomic weight, cross section (barns), number density  $\times 10^{24}$ , and the scattering mode I which is 0 for isotropic scattering on that nuclide and 1 for anisotropic scattering.

As here only a single job is to be executed, one has written -1 in the last line.

### 6.2. Interpretation of Output

The result for the sample problem given by the input data in Appendix I appears in Appendix II.

The output interpretation should follow immediately from the text given. It remains only to define the "statistical quality" Q:

$$Q = \frac{\text{Res}^2}{\sigma^2 \text{Res}}$$

where Res is the output quantity and  $\sigma^2$  the standard deviation.  $t$  is the computer time consumed (sec.). Q is substantially independent of t for a fixed computer and for a fixed number of groups. In many cases 500 or more. The standard deviation is calculated from the history plungering in groups (cf. text). The earlier MCSUP versions had a group size of 100; due to this the standard deviation was too small for the group averaging to be the normal distribution as the statistical analysis required. The present version of 500 has an improved standard deviation.

INPUT	1.1	0
-------	-----	---

### 6.3. Solutions of Various Problems by MCSUP

In the following is collected results for MCSUP calculations for a number of problems. Actually, many of these results have been produced by earlier (and less efficient) versions of the code. Due to these variable conditions, computing times and qualities are not stated.

The computations were carried out on the IBM 7094 at NEUCC, Lyngby, Denmark.

Three different resonances with the following data were treated:

Resonance no.	$E_0$	$E_u$	$E_L$	$\sigma_p$	$\Gamma_n$	$\Gamma_\gamma$
1	6.7	14	3	9.5	0.00152	0.0246
2	38.7	42	32	10.64	0.03114	0.02693
3	190.34	200	180	10.64	0.15038	0.02321

Five different moderators with the following data were treated:

Moderator	$\sigma_1$	$N_1$	$\sigma_2$	$N_2$
H	20.4	0.066633	-	-
D	3.4	0.068633	-	-
C	4.65	0.080307	-	-
H <sub>2</sub> O	20.4	0.066633	3.8	0.033317
D <sub>2</sub> O	3.4	0.066253	3.8	0.033128

(for compound moderators index 1 refers to the light component).

Six configurations with the following data were treated:

Configuration no.	a	d
1	0.2	$0.3\sqrt{\pi}$
2	1	2
3	1	$1.5\sqrt{\pi}$
4	1	$100\sqrt{\pi}$
5	1.2	$120\sqrt{\pi}$
6	1.7	$170\sqrt{\pi}$

Configurations 4, 5, 6 are, virtually, isolated fuel rods. In all the calculations to be presented, the absorber was assumed to be U-238 ( $N = 0.046974$ ,  $T = 300^\circ\text{K}$ ).

The MCSUP results are given in the table below. Concerning the scattering mode I (cf. 6.1), this refers for compound moderators to the light components: isotropic scattering is throughout assumed for the heavy component.

Resonance no.	Moderator	I	Config. no.	Resonance integral RI	Standard dev. $\sigma_{RI}^*$	Escape probability i-p	Standard dev. $\sigma_{i-p}^*$
1	H	1	1	4.512	0.014	0.11704	0.00036
1	H	0	2	1.806	0.008	0.20245	0.00094
1	H	0	3	2.7327	0.0022	0.075379	0.000058
2	H	0	3	0.9841	0.0084	0.02683	0.00023
1	H	0	5	2.740	0.009	$0.9471_{10^{-5}}$	$0.0031_{10^{-5}}$
3	H	0	3	0.1563	0.0016	0.0043018	0.000453
1	C	0	5	2.638	0.012	$0.21015_{10^{-3}}$	$0.00097_{10^{-3}}$
1	C	0	4	2.9327	0.0113	$0.23385_{10^{-3}}$	$0.00090_{10^{-3}}$
1	H	0	4	3.0319	0.0040	$0.10478_{10^{-4}}$	$0.00014_{10^{-4}}$
1	D	0	4	3.0609	0.0083	$0.87507_{10^{-4}}$	$0.00238_{10^{-4}}$
1	H	1	4	3.0793	0.0044	$0.10642_{10^{-4}}$	$0.00015_{10^{-4}}$
2	H	0	4	1.0471	0.0032	$0.36188_{10^{-5}}$	$0.00111_{10^{-5}}$
2	H	1	4	1.0557	0.0030	$0.36485_{10^{-5}}$	$0.00105_{10^{-5}}$
3	H	0	4	0.15835	0.00168	$0.54723_{10^{-6}}$	$0.00581_{10^{-6}}$
1	D <sub>2</sub> O	0	4	3.0047	0.0043	$0.79084_{10^{-4}}$	$0.00114_{10^{-4}}$
1	H	1	3	2.7639	0.0055	0.076240	0.000151
1	H	0	6	2.2492	0.0050	$0.77640_{10^{-5}}$	$0.00174_{10^{-7}}$
1	H	1	6	2.2887	0.0045	$0.79003_{10^{-5}}$	$0.00157_{10^{-7}}$
1	H <sub>2</sub> O	1	4	3.0575	0.0042	$0.10450_{10^{-4}}$	$0.00015_{10^{-4}}$
1	H <sub>2</sub> O	0	4	3.0174	0.0045	$0.10318_{10^{-4}}$	$0.00015_{10^{-4}}$

\* Later experiences indicated that these estimates of  $\sigma$  were low, in many cases by 50% or more. This underestimation was traced back to the history averaging in groups (cf. 5.6). The earlier MCSUP versions had a group size of 100; due to fluctuations induced by the negative starters this is too small for the group averages to settle down to the normal distribution as the statistical analysis requires. The present value of 500 has eliminated or greatly reduced the underestimation of  $\sigma$ .

## 7. POSSIBLE GENERALISATIONS

The MCSUP program in its present form has a number of limitations, arising from the fact that the code hitherto has served purely comparative purposes. It should, however, be possible to remove some of these limitations fairly easily, if the need occurs.

First, the lattice cell was assumed to be square, but implementation of the hexagonal geometry requires only moderate changes in the main program and in the collision probability subroutine PINCA (cf. Appendix V). The same is true for the inclusion of a cladding zone around the fuel (the RESQ-2 program<sup>3</sup>) works in hexagonal geometry with cladding).

Next, the resonance absorption has been confined to single resonances for a single absorbing nuclide type; one could perhaps desire an extension to a mixture of absorbers with quite general resonance cross sections (probably to be read in from tape). Although possible, this extension would call for fairly much editorial programming.

Inclusion of non-absorbing materials in the fuel region would, in contrast, be very easy.

Concerning the sampling method, a useful extension would be to introduce some sort of correlated sampling, which could improve the possibility of studying differential effects (e.g. the difference between isotropic and anisotropic moderator scattering). A simple way to apply correlated sampling is to force the corresponding histories in the two calculations to begin with the same random number<sup>2</sup>. In any case, correlated sampling requires a considerable extension of the statistical analysis.

## 8. SUMMARY AND CONCLUSIONS

The present report has discussed a Monte Carlo method, based on the superposition principle, devised to calculate the resonance absorption in a reactor lattice cell.

The relevant features of the corresponding FORTRAN IV computer program MCSUP (AEK P-598) have been treated, including instructions for the use of the code.

The results of a series of problems solved by MCSUP have been presented.

The experience from the work has been that the calculation efficiency proved to be quite high for most problems.

As MCSUP was intended as a reference program, it is not able to solve general resonance problems; as indicated in sec. 7, however, this restriction could probably be removed by extension of the model in various ways.

## 9. ACKNOWLEDGEMENT

The author thanks Jørn Mikkelsen who gave the incitation to much of the work. Further, Mikkelsen has reviewed part of the manuscript and has detected some errors in an earlier version of the computer code.

## 10. REFERENCES

- 1) J. Mikkelsen, Thesis work to be published as a Risø Report in 1970.
- 2) J. Spanier and E. M. Gelbard, Monte Carlo Principles and Neutron Transport Problems (Addison-Wesley, 1969).
- 3) B. L. Anderson, E. M. Gelbard, and J. Spanier, RESQ-2 - A Combined Analytic Monte Carlo Calculation of Resonance Absorption Based on Superposition. Bettis Atomic Power Laboratory, WAPD-TM-665 (March, 1967).
- 4) L. Dresner, Resonance Absorptions in Nuclear Reactors (Pergamon Press, New York, 1960).
- 5) E. M. Gelbard, L. A. Ondis II, and J. Spanier, A New Class of Monte Carlo Estimators. J. SIAM Appl. Math. 14, 697 (1966).
- 6) L. Carlvik, A method for calculating collision probabilities in general cylindrical geometry and applications to flux distributions and Dancoff factors. U.N. Int. Conf. on the Peaceful Uses of Atomic Energy, Geneva, 3. 1964. Vol. 2. Geneva 1965, p. 225.
- 7) G. N. Lance, Numerical Methods for High Speed Computers (Hire and Sons Ltd., London, 1960).
- 8) J. M. Hammersley and D. C. Handscomb, Monte Carlo Methods (Methuen, London, 1964).

- 9) S. Glasstone and M. C. Edlund, The Elements of Nuclear Reactor Theory (Van Nostrand, 1952).
- 10) A. M. Weinberg and E. P. Wigner, The Physical Theory of Neutron Chain Reactors (The University of Chicago, Chicago, 1958).
- 11) V. N. Faddeyeva and N. M. Terent'ev, Tables of values of the function

$$w(Z) = e^{-Z^2} \left( 1 + \frac{2i}{\sqrt{\pi}} \int_0^Z e^{-t^2} dt \right)$$

for complex argument (Pergamon Press, Oxford, 1961).

- 12) H. Neltrup, Error Integral with Complex Argument  $\text{erfcpl}(x, y, u, v)$ . Risø Computer Library, SA-84/1, 1965.
- 13) P. Kirkegaard, FORTRAN IV Subroutine for Error Integral with Complex Argument  $\text{ERFCPL}(X, Y, U, V)$ . Risø Computer Library, SF-84, 1970.
- 14) E. D. Cashwell and C. J. Everett, A Practical Manual of the Monte Carlo Method for Random Walk Problems. Los Alamos Scientific Laboratory, LA-2120 (1957).
- 15) K. M. Case, F. de Hoffmann, and G. Placzek, Introduction to the Theory of Neutron Diffusion, Vol. I (Los Alamos Scientific Laboratory, Los Alamos, New Mexico, 1953).
- 16) P. Kirkegaard, Nuclear Particle Transport with Emphasis on Monte-Carlo and Shielding Calculations. Danish Atomic Energy Commission, Risø Report No. 136 (1966).
- 17) D. B. MacMillan, Comparison of Statistical Estimators for Neutron Monte Carlo Calculations. Nucl. Sci. Eng. 26, 366 (1968).
- 18) M. Abramowitz and I. A. Stegun, Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables (National Bureau of Standards, Applied Mathematics Series 55, 1964).
- 19) H. Kahn, Applications of Monte Carlo. Rand. Corp., AECU-3259 (1954).
- 20) M. Leimdörfer, On the Transformation of the Transport Equation for Solving Deep Penetration Problems by the Monte Carlo Method. Trans. Chalmers Univ. Technol., Gothenburg, 266 (1964).

- 21) I. Carlvik and B. Pershagen, The Dancoff correction in various geometries. AE-16 (Stockholm, 1959).
- 22) L. W. Nordheim and G. Kuncir, A Program of Research and Calculations of Resonance Absorption. GA-2527 (1961).
- 23) D. W. Hutchinson, A New Uniform Pseudorandom Number Generator. Com. ACM 9, 432 (1966).
- 24) I. Carlvik, Dancoff Correction in Square and Hexagonal Lattices. AE-257 (Stockholm, 1966).

# IBM

[illegible]

**D 2023**

### SUPERPOSITION CALCULATION AFTER SPANIER'S SECOND METHOD

**FOUO** **UN** **NO** **2**

RANDOM INITIAL INTEGER = 55555555

NUMBER OF MODERATOR NUCLEI TYPE S- 1

RESONANCE ENERGY= 0.67000E 01 UPPER ENERGY LIMIT= 0.14000E 02 LOWER ENERGY LIMIT= 0.30000E 01

PERCENT IN SCATTERING CROSSSECTION OF ABSORBER= 0.95000E 01

GAFFA-H- 0-152005-02 GAFFA-GAFFA- 0-246005-01

APPROXIMATE WEIGHT OF ADSORBER= 0.23800E 03

TEMPERATURE OF MODERATOR = 0.90000E 03

BARRES= 0.10000E D1 PITCH= 0.26587E 01

CONCENTRATION OF ABSORBER NUCLEI = 0.46974E-01

**FILED UNDER TIME- 13:00 MINUTES**

CROSSSECTION	NUMBER DENSITY	SCATTERING MODE
+ 0.20400E 02	0.46433E-01	1

### CONTRIBUTION OF THE FIVE TERMS TO RESONANCE INTEGRAL

0-39087E-01			
0-12803E-00	DEVIATION	0-31608E-02	
0-12264E-01	DEVIATION	0-43621E-02	
0-57935E-01	DEVIATION	0-51629E-02	
0-51731E-01	DEVIATION	0-33945E-02	

0-275082-01 DEVIATION- 0-711305-02

CONFIDENTIAL PROBABILITY- 0.74134E-01 DEVIATION- 0.19421E-02

CONFIDENTIAL - SECURITY INFORMATION 199

## 13. APPENDIX III: COMPUTER CODE PRINT-OUT

```

CCPMCN/CON2/ETA,ER,A1,A2,A3,A4,SPA,CONCA
COMMON/CON3/U,V,W,SIN2,IX,TCOS(1000),TSIN(1000),AM(5),ASGPI(5),J
COMMON/CON4/PI,DIA,SM,RK,PINVS(46)
CCPMCN/CON5/ROI
DIMENSION PH(1000),SIGT(1000),SIGSR(1000),SAST(1000),TLOG(1000),
IDEG(1000,4),EQ(1000,4),RINT(5),PRES(5),DPRS(4),TDEL(4),ISMAX(2,4)
2,SMNI(5),SMNA(5),CCNCN(5),RKSN(5),ALN(5),ONALN(5),RAND(5)
3,IAHIS(5)
LOGICAL LABEL
C SPECIFICATION OF I/O UNITS
NIN=5
NOL=6
200 FORMAT(5I10)
201 FORMAT(8E10,5)
199 FORMAT(3E10,5,1I0)
WRITE(NOUT,101)
103 FORMAT(1H0,54HSUPERPOSITION CALCULATED AFTER SPANIER'S SECOND METHOD
100)
86 READ(NIN,200)IPNO
IF(IPNO.LT.0)STOP
NMAX=500
FN=NMAX
READ(NIN,200)IX,IMOD
READ(NIN,201)ER,EU,EL,SPA,GN,GG,AA,T,RC1,RK2,CCNCA,TIDMAX
DO 43 I=1,IMOD
43 READ(NIN,199)AM(I),SMNI(I),CCNCN(I),IAHIS(I)
WRITE(NOUT,103)IPNO
103 FORMAT(1H1,10HPROBLEM NO,12)
WRITE(NOUT,104)IX
104 FORMAT(1H0,23HRANDOM INITIAL INTEGER=,1I0)
WRITE(NOUT,105)IMOD
105 FORMAT(1H0,34HNUMBER OF MODERATOR NUCLEI TYPES=,12)
WRITE(NOUT,106)ER,EU,EL
106 FORMAT(1H0,17HRESONANCE ENERGY=,E12,5,3X,19HUPPER ENERGY LIMIT=,
E12,5,3X,19HLOWER ENERGY LIMIT=,E12,5)
WRITE(NOUT,107)SPA
107 FORMAT(1H0,46HPOTENTIAL SCATTERING CROSSSECTION OF ABSORBER=,E12,5
1)
WRITE(NOUT,108)GN,GG
108 FORMAT(1H0,8HGAMMA-N=,E12,5,3X,12HGAMMA-GAMMA=,E12,5)
WRITE(NOUT,109)AA
109 FORMAT(1H0,26HATOMIC WEIGHT OF ABSORBER=,E12,5)
WRITE(NOUT,110)T
110 FORMAT(1H0,28HTEMPERATURE OF MODERATOR=,E12,5)
WRITE(NOUT,111)RC1,RK2
111 FORMAT(1H0,17HRC1=,E12,5,3X,17HRK2=,E12,5)
WRITE(NOUT,112)ONALN
112 FORMAT(1H0,17HONALN=,E12,5)

```

```

WRITE(NOUT,113)TIDMAX
113 FORMAT(1H0,13HMAXIMUM TIME=,F6,2,2X,7HMINUTES)
WRITE(NOUT,114)
114 FORMAT(1H0,14HMODERATOR DATA)
WRITE(NOUT,115)
115 FORMAT(1H ,13HATOMIC WEIGHT,4X,12HCROSSSECTION,4X,14HNUMBER DENSITY
1Y,4X,15HSCATTERING MODE)
DO 44 I=1,IMOD
44 WRITE(NOUT,116)AM(I),SMNI(I),CCNCN(I),IAHIS(I)
116 FORMAT(1H ,E12,5,5X,E12,5,4X,E12,5,11X,I2)
RK=FK2/2.0
PI=2.14159265
TOP1=2.0*PI
DO 250 I1=1,1000
R=FLOAT(I1)/1000.-0.0005
ARGC=TOP1+R-PI
TCOS(I1)=COS(ARGC)
TSIN(I1)=SIN(ARGC)
250 TLOG(I1)=ALOG(R)
C=GN+GG
ETA=G*SQRT(AA/0.0013776/T/ER)
S0=2.0*CC3E6*[(AA+1.0*CC857)/AA]**2*GN/G/ER
M1=50/G*SQRT(PI)*ETA
A1=M1*GG
A2=M1*GN
A3=2.0*SQRT(50*SPA*GN*PI/G)*ETA
DELE=(EU-EL)/1000.
A4=2.0*ETA/G
DO 50 I=1,1000
M6=I
E=EL+DELE*(M6-0.5)
M7=A4*(E-ER)
CALL ERFCL(H7,ETA,U,V)
SA=A1+U*SQRT(ER/E)
SSR=SA+U+A3*V
SS=SSR+SPA
S1=SA+SS
SAST(I)=SA/ST
S1GER(I)=SSR+CONCA
50 SIGT(I)=S1*CONCA
SPAM=SPA*CONCA
CIA=2.0*ROI
R1SQ=ROI**2
AREA1=PI*R1SQ
AREA2=RK2**2-AREA1
CPAL1=4.0*AA/(AA+1.0)**2
RKSI1=6.0/(3.0*AA+2.0)
SOLR=RKSI1*SPAM*AREA1
SMH=0.0
DO 66 I=1,IMOD
ASGPI(I)=AM(I)**2+1.0
SMNA(I)=CCNCN(I)*SMNI(I)
SPR=SMH+SMNA(I)
CPAL(I)=4.0*SMNI(I)/(AM(I)+1.0)**2
ALN(I)=1.0-ONALN(I)
RKSN(I)=1.0

```



```

      IF (AM(I),GT,1.5)RKSIN(I)=1.0+ALN(I)/CHALP(I)*ALOG(ALN(I))
66 SOUR=SOUR+RKSIN(I)*SMMA(I)*AREAZ
      CALL PINCA
      IF (IMOD,EO,1)GOTO67
      RAND(I)=SMMA(I)/SHM
      DO 68 I=2,IMOD
68 RAND(I)=RAND(I-1)+SMMA(I)/SHM
67 COEF=DELE/CONCA
      DO 61 I=1,1000
      M1=I
      E=EL+DELE*(M1-0.5)
      STF=SIGT(I)
      OPT=STF*ROI
      PZERO=PO(OPT)
      PSTAR=PS(OPT)
      ASRF=ABS(STF-SPAM)
      DEG(I,1)=ASRF*(PZERO-SAST(I)*(PZERO-PSTAR))/E
      H2=SPAM*(1.-PZERO)/STF/E
      DEG(I,2)=ASRF*H2
      H3=ABS(SIGSR(I))
      DEG(I,3)=H3*H2
61 DEG(I,4)=H3*PZERO/E
      RINT(6)=SELIN(EL,EU,126,10(C.)/DELE
      PRES(6)=COEF*RINT(6)
      DO 62 L=1,4
      RINT(L)=0.
      DO 62 I=1,1000
62 RINT(L)=RINT(L)+DEG(I,L)
      CO 63 L=1,4
      J=1
      H4=0.
      H5=0.0005
      SUM=0.
      CO 54 I=1,1000
      H2=DEG(I,L)
      H3=H4
      ELN=SUM+H2
      H4=SUM/RINT(L)
65 IF (H4,LT,H5)GOTO54
      M1=I-1
      EQ(J,L)=EL+DELE*((H5-H3)/(H4-H3)+M1)
      J=J+1
      H2=PLCAT(J)/1000.-0.0005
      GCTO5E
64 CONTINUE
63 CONTINUE
      TO=0.
      J=1
      DO 382 M=1,2
      TPA=TIDMAX/16.
      GO 380 L=1,4
      IF (M,EO,2)THAX=TDEL(L)
      IE=0
      CALL TIME(ISTART)
61 REI=0.
      H=0

```

```

      IEVEN=1
      E2 H=H+1
      IF (N,EG,NMAX+1)GOTCEG
C INITIAL ENERGY
      LAEEL=.FALSE.
      IEVEN=IEVEN
      IF (IEVEN)60,60,61
60 CALL RANDL(IX,R)
      FDEC=R
      GCTCE2
61 R=1.-RODD
      E2 I1=1000.*R+1.
      E=EQ(I1,L)
      I=(E-EL)/DELE+1.
      GCTO(206,202,202,203),L
C STARTER OF TYPE 1
C INITIAL DIRECTION RELATIVE TO SURFACE NCRNAL
206 WG=SIGN(1.,SIGT(I)-SPAM)
      LAEEL=.TRUE.
207 CALL RANDL(IX,R)
      U1=SQRT(R)
      M1=SQRT(1.-R)
      CALL RANDL(IX,R)
      I1=1000.*R+1.
      V=H1*TSIN(I1)
      SIN2=1.-V**2
C REJECTION CALCULATION
      H2=DEXP(DIA*U1/SIN2+SIGT(I))
      CALL RANDL(IX,R)
      IF (H2,LE,R)GOTO207
      V1=H1*TCOS(I1)
C INITIAL SURFACE POSITION AND CORRECTION OF DIRECTION
      CALL RANDL(IX,R)
      I1=1000.*R+1.
      COSV=TCOS(I1)
      SINV=TSIN(I1)
      X=FOI+COSV
      Y=FCI+SINV
      U=COSV*U1-SINV*V1
      V=SINV*U1+COSV*V1
      K=1
      GCTO5
C STARTER OF TYPE 2 OR 2
C INITIAL POSITION IN ZONE 1
202 CALL RANDL(IX,R)
      RCSC=R*RSO
      RC=SQRT(RSOQ)
      CALL RANDL(IX,R)
      I1=1000.*R+1.
      H=RC*TCOS(I1)
      Y=FCI*TSIN(I1)
      CALL RSO
C INITIAL PATH LENGTH IN ZONE 1
      CALL RANDL(IX,R)
      I1=1000.*R+1.
      RL=-TLOG(I1)/SIGT(I)

```

## C REJECTION CALCULATION

```

X=X+U*RL
Y=Y+V*RL
RDSQ=X**2+Y**2
IF (RDSQ.GT.R1SQ) GOTO202
K=1
IF (L.EQ.3) GOTO204
WG=SIGN(1.,SIGT(I))-SPAW
GOTO15
204 WG=SIGN(1.,SIGSR(I))
GOTO15

```

## C STARTER OF TYPE 4

## C INITIAL DIRECTION RELATIVE TO SURFACE NORMAL

```

203 CALL RANDL(IX,R)
U1=-SQRT(R)
V1=SQRT(1.-R)
CALL RANDL(IX,R)
I1=1000.*R+1.
V=H1*TSIN(I1)
SIN2=1.-V**2

```

## C REJECTION CALCULATION

```

H2=-C1*U1/SIN2*SIGT(I)
CALL RANDL(IX,R)
IF (CEXP(H2).LE.R) GOTO203
V1=H1*TCOS(I1)

```

## C INITIAL SURFACE POSITION AND CORRECTION OF DIRECTION

```

CALL RANDL(IX,R)
I1=1000.*R+1.
COSV=TCOS(I1)
SINV=TSIN(I1)
X=RD1+COSV
Y=RD1+SINV
U=COSV+U1-SINV*V1
V=SINV+U1+COSV*V1

```

## C COVEYOU SAMPLING OF INITIAL RESONANCE SCATTERING IN ZONE 1

```

CALL RANDL(IX,R)
I1=1000.*R+1.
H1=-TLOG(I1)
I1=H1/H2
H3=I1
RL=(H1-H3*H2)/SIGT(I)

```

## C POSITION OF INITIAL SCATTERING

```

X=X+U*RL
Y=Y+V*RL
RDSQ=X**2+Y**2
WG=SIGN(1.,SIGSR(I))
K=1

```

## C NOW BEGINS COMMON HISTORY BLOCK

## C SAMPLING OF ENERGY AFTER SCATTERING IN ZONE 1

```

11 CALL RANDL(IX,R)
E=E*(1.-RDSQ*AL1)
IF (E.LT.EL) GOTO82
I=(E-EL)/DELE*1.

```

## C ISOTROPIC DIRECTION ZONE 1

```

CALL ISO

```

## C PATH LENGTH IN ZONE 1

```

6 CALL RANDL(IX,R)

```

```

I1=1000.*R+1.

```

```

RL=-TLOG(I1)/SIGT(I)

```

## C DISTANCE TO ZONE INTERFACE

```

DELTA=U*X+V*Y

```

```

IF (K.EQ.2) GOTO13

```

```

SL=(SQRT(DELTA**2+SIN2*(R1SQ-RDSQ))-DELTA)/SIN2
GCTC14

```

```

13 K=1

```

```

SL=-2.*DELTA/SIN2

```

```

14 IF (SL.CE.RL) GOTO4

```

```

X=X+U*SL

```

```

Y=Y+V*SL

```

## C MODIFIED TRACK LENGTH SCORING IN CASE OF PASSAGE

```

RES=RES+WG*SAST(I)*SIGT(I)*SL

```

```

GOTO5

```

## C COLLISION IN ZONE 1

```

4 CALL RANDL(IX,R)

```

```

IF (R-SAST(I))16,17,17

```

## C ABSORPTION

```

16 IF (LABEL) GOTO207

```

## C MODIFIED TRACK LENGTH SCORING IN CASE OF ABSORPTION

```

H1=SL*SIGT(I)

```

```

I2=OEXP(H1)

```

```

RES=RES+WG*SAST(I)*(1.-(1.-H2)*(H1+1.))/H2
GOTO62

```

## C SCATTERING IN ZONE 1

```

17 IF (LABEL) GOTO19

```

## C MODIFIED TRACK LENGTH SCORING IN CASE OF SCATTERING

```

RES=RES+WG*SAST(I)*SIGT(I)*RL

```

```

19 X=X+U*RL

```

```

Y=Y+V*RL

```

```

RDSQ=X**2+Y**2

```

```

GCTC15

```

## C PATH LENGTH IN ZONE 2

```

5 CALL RANDL(IX,R)

```

```

I1=1000.*R+1.

```

```

RL=-TLOG(I1)/SHM

```

```

IF (K.EQ.1) GOTO7

```

```

DELTA=U*X+V*Y

```

```

IF (DELTA.GT.0.) GOTO7

```

```

8 D=DELTA**2-SIN2*(RCSQ-R1SQ)

```

```

IF (D.LT.0.) GOTO7

```

## C DIRECTION TOWARDS ZONE 1

```

T=(-DELTA-SQRT(D))/SIN2

```

```

IF (T.GT.RL) GOTO9

```

```

X=X+U*T

```

```

Y=Y+V*T

```

```

GCTC6

```

## C DIRECTION TOWARDS OUTER BOUNDARY

```

7 K=2

```

```

F1=SIGN(RK,U)

```

```

TX=(H1-X)/U

```

```

F2=SIGN(RK,V)

```

```

TY=(H1-Y)/V

```

```

T=TY

```

```

      IF (TX,LT,TY) T=TX
      IF (TX,GT,RL) GOT09
C REFLECTION
      X=X+U*T
      Y=Y+V*T
      ROSQ=X**2+Y**2
      RL=RL-T
      IF (TX-TY) 10,11,11
10  L=-U
      GCTC12
11  V=-V
12  DELTA=U*X+V*Y
      IF (DELTA) 8,8,7
C ENERGY SAMPLING FOR ZONE 2 COLLISION
9  CALL RANDL(IX,R)
      IF (INCO,EO,1) GOT065
      DO 70 J=1,IMOD
70  IF (R,LT,RAND(J)) GOT071
71  H1=0.
      IF (J,GT,1) H1=Rand(J-1)
      R=(R-H1)/(RAND(J)-H1)
69  E=E*(1.-R*ONALM(J))
      IF (E,LT,EL) GOT082
      I=(E-EL)/DELE+1.
C POSITION FOR ZONE 2 COLLISION
      LABEL=,FALSE.
      X=X+U*RL
      Y=Y+V*RL
      ROSQ=X**2+Y**2
C ISOTROPIC DIRECTION ZONE 2
      IF (IANIS(J),EO,1) GOT072
      CALL ISO
      GOTC5
72  CALL ANIS
      GOT08
80  IS=IS+1
      PH(IS)=RES
      CALL TIME(ISLUT)
      TID=FLQAT(ISLUT-1START)/3600.
      IF (TID,LT,TMAX,AND,IS,LT,1000) GOT081
      TC=TID*TID
      H1=0.
      H2=0.
      DO 85 I=1,IS
      H2=PH(I)
      H1=H1+H2
85  P2=H2+H2**2
      R1=IS
      H1=H1/RS
      H2=H2/RS
      H3=CCOFPRINT(L)/RN
      H4=H1+H3
      H5=SQRT((H2-H1**2)/(R2-1.))*H3
      ISMAX(H5,L)=IS
      IF (P,EO,1) GOT087
      H1=ISMAX(1,L)

```

```

      H2=ISMAX(2,L)
      H3=H1+H2
      PRES(L)=(H1*PRES(L)+H2*H4)/P3
      DPRS(L)=SQRT((H1*DPRS(L)**2+(H2*H5)**2)/H3)
      GCT0350
87  PRES(L)=H4
      DPRS(L)=H5
350  CCNTINUE
      IF (H,EO,2) GOT0352
      H1=0.
      DO 353 L=1,4
353  H1=H1+DPRS(L)
      DO 354 L=1,4
      TDEL(L)=TIDMAX*(DPRS(L)/H1-0.625)
354  IF (TDEL(L),LT,0.2) TDEL(L)=0.2
352  CCNTINUE
      P=PRES(5)-PRES(1)-PRES(2)+PRES(3)+PRES(4)
      H1=0.
      DO 351 L=1,4
351  H1=H1+DPRS(L)**2
      CP=SQRT(H1)
      TC=(P/DP)**2/TD/60.+0.5
      WRITE(NOUT,121)
121  FCFMAT(1H0,52HCONTRIBUTION OF THE FIVE TERMS TO RESONANCE INTEGRAL
1)
      WRITE(NOUT,122)PRES(5)
122  FCFMAT(1H,37HDETERMINISTIC TERM=,E12.5)
      WRITE(NOUT,123)PRES(1),DPRS(1)
123  FCFMAT(1H,37HNEGATIVE OUTWARD SURFACE SOURCE TERM=,E12.5,3X,
110HDEVIATION=,E12.5)
      WRITE(NOUT,124)PRES(2),DPRS(2)
124  FCFMAT(1H,37HNEGATIVE VOLUME SOURCE TERM=,E12.5,3X,
110HDEVIATION=,E12.5)
      WRITE(NOUT,117)PRES(3),DPRS(3)
117  FCFMAT(1H,37HPOSITIVE VOLUME SOURCE TERM=,E12.5,3X,
110HDEVIATION=,E12.5)
      WRITE(NOUT,118)PRES(4),DPRS(4)
118  FCFMAT(1H,37HPOSITIVE INWARD SURFACE SOURCE TERM=,E12.5,3X,
110HDEVIATION=,E12.5)
      WRITE(NOUT,119)P,DP
119  FCFMAT(1H0,37HRESONANCE INTEGRAL=,E12.5,3X,
110HDEVIATION=,E12.5)
      H1=AREAL+CCNCA/SOUR
      P=H1*P
      CP=H1*CP
      WRITE(NOUT,120)P,DP
120  FCFMAT(1H0,37HRESONANCE CAPTURE PROBABILITY=,E12.5,3X,
110HDEVIATION=,E12.5)
      WRITE(NOUT,125)IQ
125  FCFMAT(4H0,20HSTATISTICAL QUALITY=,IS)
      GCT0350
      END

```

```

SLEFCLTIME ISN
CCPMCN/CCN3/U,V,W,SIN2,IX,TCCS(1000),TSIN(1000),AN(5),ASCPL1(5),J
CALL RANDL(IX,R)
V=1.*R-1.
SIN2=1.-W**2
F1=SQRT(SIN2)
CALL RANDL(IX,R)
I1=(00.*R+1.
L=F1*TCOS(I1)
V=F1*TSIN(I1)
RETURN
END

```

```

SUEFCLTIME ANIS
CCPMCN/COM3/U,V,W,SIN2,IX,TCCS(1000),TSIN(1000),AN(5),ASCPL1(5),J
CALL RANDL(IX,R)
IF (AN(J).LT.1.E)GOTC1
RMV=2.*R-1.
H1=AN(J)*RMV
A=(1.+H1)/SQRT(ASCPL1(J)+2.*H1)
E=SQRT(1.-A**2)
GOTO2
1 A=SQRT(R)
B=SQRT(1.-R)
2 CALL RANDL(IX,R)
I1=1000.*R+1.
C=TCOS(I1)
D=TSIN(I1)
EC=E*C+W
BC=B*C
SUIN=SQRT(SIN2)
UEF=(BC*U-BD*V)/SUIN+A*U
V=(BC*V+BD*U)/SUIN+A*V
L=UEF
B=U*W-E*C+SUIN
SIN2=1.-W**2
RETURN
END

```

```

FUNCTION PS(T)
CCPMCN/COM4/PI,DIA,SHW,RK,PINVS(46)
IF (T.GE.0.)GOTO1
IF (T.GE.1.)GOTC2
A=20.
B=1.
GOTO3
2 A=0.
B=10.
3 F1=A+T+B
I1=F1
F2=PINVS(I1)
F1=FINVS(I1+1)
F4=F1-F2
H8=F1
V6=F1-H8
H7=H8+V6*H4
F5=1./H7
RETURN
1 PI=C./PINVS(46)/T
RETURN
END

```

```

SLEFCLTIME PINA
CCPMCN/COM4/PI,DIA,SHW,RK,PINVS(46)
CCPMCN/CCN5/RO1
DIMENSION X1(8),Y1(8),W1(8),L1(64),V1(64),ST1(10),PESC(10),
1RESULT(10),OPTCH(10),CPT1(10),OPT2(10),CAM(10),RESNO(10),
2STMCH(10),PSTAR1(46),YG(2),W61(8)
DATA YG,W61/0.001992,,020E42,,02226E,,2C6855,,394469,,617311,
1.225E3,,984756,,003295,,017R43,,045439,,079220,,106047,,112586,
2.051119,,044551/
CF1FO=RO1*SHW
CFR1SQ=CPTRO**2
OPTRK=RK*SHW
DO 301 I=1,8
Y1(I)=OPTFO*(1.-YG(I))
301 X1(I)=SQRT(OPR1SQ-Y1(I)**2)
CA=PI/126.
CC 302 J=1,32
ARCU=CA*(FLOAT(J)-C.E)
H1=CCS(ARGU)
F2=SIN(ARGU)
U1(J)=H1
V1(J)=H2
J2=65-J
U1(J2)=H1
302 V1(J2)=H2
DO 303 K=1,10
F1=1.2*FLOAT(K)
STMCH(K)=H1
ST1(K)=H1/DIA
PESC(K)=PO(H1/2.)
RESNO(K)=0.
303 RESULT(K)=0.
CC 350 J=1,6
X0=X1(I)
Y0=Y1(I)
W0=W61(I)
F1=2.*X0/SHW
DO 304 K=1,10
F2=-1*ST1(K)
OPTCH(K)=H2
304 RESNCH(K)=RESNCH(K)+8IC3(H2)*W0
CC 350 J=1,64
U=U1(J)
V=Y1(J)
X=X0*U-Y0*V
Y=X0*V+Y0*U
DO 305 K=1,10
CPT1(K)=0.
305 OPT2(K)=OPTCH(K)
306 CFTMCD=0.
307 TX=(OPTRK-X)/U
H1=SIGN(OPTRK,V)
TY=(H1-Y)/V
IF (TY.LT.TX)GOTO311
CF1MOD=OPTMOD+TX
X=-CPTRK
Y=Y+V*TX
GOTO312
311 CFTMCD=OPTMOD+TY
X=X+U*TY
Y=Y+V*TY
312 C=CFR1SQ-(U*Y-V*X)**2
IF (C)307,313,313
313 T=U*Y-V*X-SQRT(D)

```

```

CFYMOD=OPTMOD+T
CC 308 K=1,10
CP1AC=OPT2(K)
CPTBC=OPT1(K)
OPTAD=OPTAC+OPTMOD
CPTBC=OPTBC+OPTMOD
H1=BIC3(OPTBC)-BIC3(OPTAC)-BIC3(OPTED)+BIC3(OPTAD)
308 RESULT(K)=RESULT(K)+H1*NG
Y=Y+U+T
V=Y+V+T
DELTA=U*X+V*Y
SL=-2.*DELTA
X=X+U*SL
Y=Y+V*SL
SL=SL/SHM
IF(OPT1(1)+SL*ST1(1)+CPTMCD,GT,7.)GOTO350
CC 309 K=1,10
H1=SL*ST1(K)+CPTMOD
CPT1(K)=OPT1(K)+H1
309 CPT2(K)=OPT2(K)+H1
GOTO308
30C CONTINUE
H2=PI/8.
CC 311 K=1,10
351 RESULT(K)=RESULT(K)+PESC(K)/64./{H2-RESNCH(K)}
PSTAR1(1)=1.
PSTAR1(13)=RESULT(1)
CC 346 K=2,10
L=2*K+16
346 PSTAR1(L)=RESULT(K)
DO 347 K=1,10
X=ETPMCH(K)
P=PESC(K)
PST=RESULT(K)
H1=X*P*PST
347 GAM(K)=H1/(P-PST+H1)
GAMMA=GAM(1)
CC 348 L=2,12
H1=L
X=(H1-1.)/1C.
P=PQ(X/2.)
348 PSTAR1(L)=P*GAMMA/{1.-(1.-GAMMA)*(1.-X*F)}
H2=GAM(1)
H3=GAM(2)-H2
CC 349 L=14,21
H1=L
X=(H1-1.)/10.
GAMMA=H2+H3/1.2*(X-1.2)
P=PQ(X/2.)
349 PSTAR1(L)=P*GAMMA/{1.-(1.-GAMMA)*(1.-X*F)}
CC 260 K=3,10
H1=K
X1=1.2*H1-1.2
H3=GAM(K-1)
H4=GAM(K)-H3
CC 360 M=1,2
H2=M
X=X1+C.4*H2
GAMMA=H3+H4/3.0*H2
P=PQ(H/2.)
L=2*K+H+13
360 PSTAR1(L)=P*GAMMA/{1.-(1.-GAMMA)*(1.-X*F)}
CC 261 L=1,40
361 PSTAR1(L)=L./PSTAR1(L)

```

- 37 -

```

FUNCTION SELIN(A,B,T,RN)
COMMON/CON1/XX,YY
DIMENSION FUNC(129)
F=(E-A)/FLOAT(N)
NF=N+1
DO 11 I=1,NP1
XX=A+FLOAT(I-1)*H
CALL FCT
11 FUNC(I)=YY
DEL=C.
CC 1 I=1,N
1 CEL=DEL+ABS(FUNC(I+1)-FUNC(I))
RI=C.
CC 2 I=1,N
XL=A+FLOAT(I-1)*H
HJ=FM*ABS(FUNC(I+1)-FUNC(I))/DEL+1.
HJ2=HJ/FLOAT(HJ)
HJ=HJ2/2.
RIJ=FUNC(I)+FUNC(I+1)
XX=XL-HJ
H3=0.
CC 3 J=1,MJ
XX=XX+HJ2
CALL FCT
3 H1=H1+YY
RIJ=RIJ+4.*H1
IF(HJ.EQ.1)GOTO2
XX=XL
MJ=MJ-1
H2=0.
CC 4 J=1,MJ
XX=XX+HJ2
CALL FCT
4 H2=H2+YY
RIJ=RIJ+2.*H2
2 FI=RI+RIJ*HJ/3.
SELIN=FI
RETURN
END

```

```

SLEROLTIME FCT
COMMON/CON1/XX,YY
COMMON/CON2/ETA,ER,A1,A2,A3,A4,SPA,CENCA
COMMON/CON3/RO1
H7=A4*(XX-ER)
CALL ERFCLP(H7,ETA,U,V)
SA=A1+U*SQRT(ER/XX)
E5=A2*U+A3*V+SPA
ST=SA+SS
CPI=ST+CENCA*RC1
PSTAR=PS(OP7)
YY=SA*CONCA*(PSTAR+SPA/ST*(1.-PSTAR))/XX
RETURN
END

```

```

B=C*0.1E4273203/(A**2+P1)
L=L+B
V=Y+A*B
A=X+2.275E57C805+T
E=C*(0.1243124432/(A**2+P1)
U=U+B
V=Y+A*B
A=X+2.020E3702E1+T
B=0.(0C02725CE5347/(A**2+P1)
L=L+B
V=Y+A*B
A=X+2.EE97248579+T
B=0.(0C0000E462432E41/(A**2+P1)
U=L+B
V=Y+A*B
IF (T.LT.O.)GOTO6
T=-T.
GOTO5
U=U*Y
4 V=Y*FLOAT(N)
IF (C.EQ.O..AND.O.EQ.O.)RETURN
C=EXP(-C)
U=2.*(C*COS(D)-U
V=S.*(C*SIN(D)-V
RETURN
END

```

[illegible]

**TURN**      The above is typical of Colloids in a New York reservoir; the  
**TURN**      volume of 1/l. at  $\phi = 1.4$ , 2.1, and 3.0 showed values 5.188, 4.984,  
**D**      and 7.959.



15. APPENDIX V: FUEL-TO-MODERATOR COLLISION PROBABILITY  $P^*$ 

We now turn to discuss the calculation of the fuel-to-moderator collision probability  $P^*$ , which occurs in (31) and (32).

Also here we shall follow Spanier and Gelbard<sup>2)</sup>, but only to a certain extent; we found it much more satisfactory to compute  $P^*$  by purely deterministic methods instead of using Monte Carlo.

$P^*(E)$  is defined as the probability that a source neutron with energy  $E$  drawn from a uniform distribution in the fuel will make its first collision in the moderator.

As the moderator cross section is assumed energy independent, we first notice that  $P^*$  for a fixed configuration is a function only of the optical radius  $\zeta = a \Sigma$  of the fuel.

As for  $P_0$  (Appendix IV) we want to construct a table of  $P^*(\zeta)$ , or rather  $1/P^*(\zeta)$ , this quantity behaving more linearly. This table is of course not universal but depends on the actual configuration. This table has (like the  $1/P_0$ -table) 46 entries with  $\zeta = 0, 0.05, 0.1, \dots, 0.95, 1.0, 1.2, \dots, 5.8, 6.0$ , and linear interpolation should be quite adequate to find  $P^*(\zeta)$  for arbitrary  $\zeta$ . Now Spanier and Gelbard<sup>2)</sup> point out that it is not necessary to calculate all the corresponding 46  $P^*$ -values. They take advantage of Nordheim's approximate formula<sup>22)</sup>

$$P^*(\zeta) \approx \frac{P_0(\zeta) G_m}{1 - (1 - G_m)(1 - G_f(\zeta))} \quad (50)$$

where  $G_m$  and  $G_f(\zeta)$  is the sticking probability<sup>2, 22)</sup> for the moderator and fuel, respectively, and define a quantity  $\gamma_m(\zeta)$  which exactly satisfies the following equation

$$P^*(\zeta) = \frac{P_0(\zeta) \gamma_m(\zeta)}{1 - (1 - \gamma_m(\zeta))(1 - G_f(\zeta))} \quad (51)$$

Due to (50),  $\gamma_m(\zeta)$  is a slowly varying function of  $\zeta$ . Now,  $G_f(\zeta)$  is given by<sup>2)</sup>

$$G_f(\zeta) = 2\zeta P_0(\zeta) \quad (52)$$

so, assuming  $P_0(\zeta)$  to be known,  $\gamma_m(\zeta)$  becomes a unique function of  $P^*(\zeta)$ . The idea is now to compute (in some way)  $P^*(\zeta)$  at rather few  $\zeta$ -values, e.g.  $\zeta = 0.6, 1.2, \dots, 6.0$ . Via (51) one obtains 10 corresponding  $\gamma_m$ -values, and linear interpolation in this coarse  $\gamma_m$ -table renders next a full, 46-entry table for  $\gamma_m(\zeta)$ , which finally, again via (51), is converted to the desired 46-entry table for  $P^*(\zeta)$ .

Now it still remains to calculate  $P^*(\zeta)$  at  $\zeta = 0.6, 1.2, \dots, 6.0$ , and in doing so, we now digress from the exposition of Spanier and Gelbard<sup>2)</sup>, who recommend a Monte Carlo procedure for this job. We have found that an approach, inspired by Carlvik<sup>6, 24)</sup>, yield a both faster and more accurate calculation scheme.

Consider fig. 3 which shows a part of the infinite lattice system.  $P^*$  may be expressed by the double integral

$$P^* = \frac{2}{\pi} \int_{-\pi/4}^{\pi/4} d\alpha \int_0^a dy l(y) \frac{1}{2\pi a^2} p(y, \alpha) \quad (53)$$

where  $p(y, \alpha)$  is the probability that a starter drawn uniformly from the chord of length  $l(y)$  after travelling in the direction  $\alpha$  makes its first collision in the moderator. Referring to the optical scale on the  $\alpha$ -ray depicted in fig. 3 below, one can show that  $p(y, \alpha)$  is given by the expression

$$p(y, \alpha) = \frac{\sum_{i=1}^{\infty} \left\{ \left[ K_{1/3}(t_i) - K_{1/3}(t_i + \lambda) \right] - \left[ K_{1/3}(u_i) - K_{1/3}(u_i + \lambda) \right] \right\}}{\Sigma \cdot l(y)} = \frac{p(y, \alpha)}{\Sigma \cdot l(y)} \quad (54)$$

where  $K_{1/3}(x)$  is the Bickley function

\* Mikkelsen<sup>1)</sup> has noticed a pronounced maximum in the relative error of (50) in the range  $0 < \zeta < 0.6$  for small cells, perhaps making the above procedure questionable for such cells. We have, however, not revealed any significant errors for the final calculation results.



$$K_{In}(x) = \int_0^{\frac{\pi}{2}} \exp\left(-\frac{x}{\cos\theta}\right) \cos^{n-1} \theta d\theta \quad (55)$$

Then (53) may be written

$$P^* = \frac{4}{\pi^2 \lambda a^2} \int_{-\frac{\pi}{4}}^{\frac{\pi}{4}} da \int_0^a dy \varphi(y, a) \quad (56)$$

In the numerical calculation of the double integral (56), a discrete representation is chosen for both variables:

- 1)  $a$ -integration: 64 equidistant  $a$ -values in  $-\frac{\pi}{4} < a < \frac{\pi}{4}$
- 2)  $y$ -integration: 8 non-equidistant  $y$ -values in  $0 < y < a$ .

The  $y$ -integration is of the Gaussian type. We choose a weight function mimicking the behaviour of the integrand near  $y = 0$ . For small  $y$ ,  $\varphi(y, a)$  tends to be proportional to  $1(y)$  and thereby to  $y^{1/2}$ . Hence, by introducing the new variable  $x = (y/a)^2$ , one could hope that the Gaussian representation

$$\int_0^1 x f(x) dx = \sum_{i=1}^n w_i f(x_i) \quad \text{with} \quad \sum_{i=1}^n w_i = \frac{1}{2} \quad (57)$$

would yield a good approximation to the integral in question. Ref. 18 gives tables of the  $x_i$  and  $w_i$  for different choices of  $n$ . With such a set of  $(x_i, w_i)$  one would then write

$$\int_0^a dy \varphi(y, a) \approx 2a \sum_{i=1}^n w_i \varphi(a x_i, a) \quad (58)$$

where  $x_i = x_i^2$ . In fact, (58) has proved to be a surprisingly accurate approximation, and it seems rather difficult to justify mathematically the success of this procedure, which has been used widely in collision prob-

ability problems\*. In our case,  $n = 8$  was found to give an adequate compromise between speed and precision.

Hence, (56) is replaced by

$$P^* \approx \frac{1}{16\pi\lambda a} \sum_{i=1}^8 w_i \sum_{j=1}^{64} \varphi(a\eta_i, a_j) \quad (59)$$

where  $\eta_i$  and  $w_i$  are given below

i	$\eta_i$	$w_i$
1	0.001992	0.003295
2	0.020842	0.017843
3	0.082268	0.045439
4	0.208655	0.078200
5	0.394469	0.106047
6	0.617311	0.112506
7	0.825693	0.091119
8	0.984756	0.044551

We go a step further in refining the formula for  $P^*$  by reducing the discretization error. We observe that an expression quite similar to (59) exists for  $P_0$ :

$$P_0 \approx \frac{1}{16\pi\lambda a} \sum_{i=1}^8 w_i \sum_{j=1}^{64} \varphi_0(a\eta_i, a_j) \quad (60)$$

where  $\varphi_0$  is the function  $\varphi(y, a)$  appropriate for an isolated fuel rod; (54) gives

\* The method should presumably be credited to H. Tollander and I Carlvik, Sweden.

$$\varphi_0(y, a) = Ki_3(0) - Ki_3(\tau) = \frac{\pi}{4} - Ki_3(\tau) \quad (61)$$

As we already have an efficient algorithm for  $P_0$  to our disposal we recast (59) to the following form

$$P^* \approx \frac{\sum_{i=1}^8 w_i \sum_{j=1}^{64} \varphi(a_i, a_j)}{\sum_{i=1}^8 w_i \sum_{j=1}^{64} \varphi_0(a_i, a_j)} \cdot P_0 \quad (62)$$

Formula (62) has the further advantage of yielding the exact limit  $P_0$  when the cell expands towards the isolated fuel rod.

The evaluation of the  $Ki_3$ -functions is performed by the function subprogram BIC3, developed originally by B. Tollander in connection with Carlvik's work<sup>6)</sup>. This procedure zeros  $Ki_3(x)$  for  $x > 10$ , and this leads to a natural truncation of the infinite series in (54).

The calculation technique suggested in this Appendix has been programmed as to form the subroutine PINCA. PINCA delivers the 48  $P^*$ -values discussed earlier. Computation of  $P^*(\zeta)$  by linear interpolation in the  $1/P^*$ -table so constructed is done by another subprogram, PS. For  $\zeta \geq 6$  PS uses the approximation<sup>2)</sup>

$$P^*(\zeta) \approx P^*(6) \cdot \frac{6}{\zeta} \quad (63)$$

## 16. APPENDIX VI: A SELECTIVE INTEGRATION ROUTINE

The evaluation of the integral (31) is carried out by a special, selective integration routine SELIN. This was found useful because the integrand in (31) varies rather sharply around the resonance energy; as one wishes to limit the number  $N$  of integrand computations (here,  $N = 1000$  was chosen) it is important to make a selective choice of the corresponding  $N$  abscissas.

SELIN does this in a crude, yet adequate manner. It works roughly in such a way that it divides the integration interval in a coarse equidistant subdivision. Thereafter, each of these subintervals  $\Delta_i$  is submitted to a further subdivision in  $m_i$  new subintervals  $\delta_{ij}$ ,  $m_i$  being roughly proportional to the derivative of the integrand at  $\Delta_i$ . Finally, Simpson's rule is applied to all subintervals and the results added. SELIN is in no way an "optimal" routine, but has nevertheless proved useful in the calculation of various definite integrals with a pronounced variation of the integrand.

SELIN is here supplied with function values from the function subprogram FCT.

## 17. APPENDIX VII: A FAST EXPONENTIAL ROUTINE

During the Monte Carlo calculations performed by MCSUP it is frequently required to calculate the exponential  $e^{-x}$ , or, more often, the function  $1 - e^{-x}$ . Of course the standard exponential function available at the computer may be applied, but in view of the rather limited demands for precision in Monte Carlo calculations it is better policy to use a faster routine and sacrifice some accuracy. Spanier and Gelbard<sup>2)</sup> recommend the use of the so-called  $g$ -function defined by

$$g(x) = \frac{x}{1 - e^{-x}} \quad (64)$$

After tabulating  $g(x)$  from e. g. 0 to 7 with steps of 0.2 it is possible by linear interpolation to calculate  $g(x)$  to an accuracy within 0.1% for  $x > 7$  they set  $1 - e^{-x}$  equal to one.

Another alternative is to construct a simple polynomial approximation of  $1 - e^{-x}$ , and this method performed quite well on the IBM 7094 and was

$x = 0.1$	$g(x) = 1.0$	total polynomial
$x = 0.2$	0	error zero

adopted in MCSUP as the function subprogram OEXP. It was decided to divide the interval  $0 < x < 8$  in four intervals in such a manner that third order polynomials chosen in the best possible way would yield absolute errors of almost the same magnitude in the four intervals. "Best possible" is understood in the Chebyshev sense, i.e. one minimizes the max. absolute error in each interval. The relevant approximation formula is given by Lance <sup>7)</sup>:

$$\exp(kx) = I_0(k) + 2 \sum_{n=1}^{\infty} I_n(k) T_n(x), \quad -1 \leq x \leq 1 \quad (65)$$

where  $I_n$  is the modified Bessel function of first kind and order  $n$ , while  $T_n(x)$  is the Chebyshev polynomial of degree  $n$ . By truncating (65) at  $T_3$  and making a variable transformation, this formula yields the polynomial coefficients. After some numerical experimentation the interval subdivision and coefficients stated in OEXP (Appendix III) was obtained. The OEXP approximations are not quite those of the Chebyshev type, because small constant terms had been added in order to keep the mean arithmetic error in the whole range  $0 < x < 8$  as low as possible, whereby  $\int_0^8 e^{-x} dx$  calculated by OEXP becomes 1 with a high degree of precision (OEXP sets  $1 - e^{-x}$  equal to one for  $x > 8$ ).

The performance of the g-function method (G) and OEXP has been measured on an IBM 7094 (model I) with the result shown in the table below.

	G	OEXP
netto time ( $\mu$ S) for a single call	240	180
root-mean-square error	$1.10 \cdot 10^{-4}$	$1.27 \cdot 10^{-4}$
max. error	$3.01 \cdot 10^{-4}$	$2.37 \cdot 10^{-4}$
abscissa for max. error	0.90	0.98
mean arithmetic error	$-0.73 \cdot 10^{-4}$	$10^{-6}$
mean absolute error	$0.73 \cdot 10^{-4}$	$1.13 \cdot 10^{-4}$
truncation error	$9.1 \cdot 10^{-4}$	$3.35 \cdot 10^{-4}$
zero value	0	$1.93 \cdot 10^{-4}$

The time for a call of the IBM 7094 standard function  $1 - e^{-x}$  was found to 262  $\mu$ S.

#### 18. APPENDIX VIII: OPTIMIZATION OF COMPUTER TIME FOR THE FOUR TYPES OF STARTERS

The resonance absorption rate was in sec. 3 expressed as the sum of 5 terms (eq. (40)), where one is deterministic and the remaining four are Monte Carlo terms. The combined Monte Carlo expression

$$R = -I_1 - I_2 + I_3 + I_4 = R_1 + R_2 + R_3 + R_4 \quad (66)$$

was composed of four mutually independent terms to be calculated separately according to the principles laid down in sec. 3. Now, some possibilities exist for carrying out a more refined composition of the terms in (66). Spanier and Gelbard <sup>2)</sup> propose a sampling technique, where the initial energy  $E$  is sampled from a combined density function, derived from the four individual density functions, whereafter the category of starter is selected. This procedure implies saving of computer storage.

In our case, small computing time rather than storage saving is at a premium, and then another approach is feasible.

We still calculate the four terms in (66) separately, but we assume that a certain amount of computer time  $T$  is at our disposal and try to find a division

$$T = T_1 + T_2 + T_3 + T_4 \quad (67)$$

where  $T_1$  is the time spent in calculation of  $R_1$ , so that a minimum-variance estimate of  $R$  is obtained. We have

$$\sigma^2(R) = \sum_{i=1}^4 \sigma^2(R_i) \quad (68)$$

At this point we assume that

$$\sigma^2(R_i) = \frac{k_i}{T_i} \quad (69)$$

where the  $k_i$  are known positive constants. Hence we minimize the functional

$$\varphi(T_1, T_2, T_3, T_4) = \sum_{i=1}^4 \frac{k_i}{T_i} \quad (70)$$

under the constraint (67). The method of Lagrange's multipliers yields easily the solution

$$T_i = \frac{\sqrt{k_i}}{\sum_{i=1}^4 \sqrt{k_i}} T \quad (71)$$

In practice we have to estimate the  $k_i$ . This is done in MCSUP by applying a preliminary computation with time length  $T'$  distributed equally with  $T_i' = \frac{T'}{4}$  to each type of starters. Preliminary variance estimates

$$\sigma_i'^2 \approx \sigma^2(R_i') \quad (72)$$

are used to calculate  $k_i$  as

$$k_i = T_i' \sigma_i'^2 \quad (73)$$

Now (71) is recast to

$$T_i = \frac{\sigma_i'}{\sum_{i=1}^4 \sigma_i'} \cdot T \quad (74)$$

Hereafter the main calculation is executed. Utilizing also the preliminary calculation, we have

$$T_i'' = T_i - T_i' \quad (75)$$

where  $T_i''$  is the time spent in calculation of  $R_i$  in the main calculation. The MCSUP program uses  $T' = \frac{T}{4}$  and with this choice we finally obtain

$$T_i'' = \left( \frac{\sigma_i'}{\sum_{i=1}^4 \sigma_i'} - \frac{1}{16} \right) T \quad (76)$$

## 19. APPENDIX IX: COMPUTER DEPENDENT FEATURES OF MCSUP

MCSUP in its present form is intended for use on the IBM 7094 at the NEUCC center in Lyngby, Denmark. In view of the possibility of its use on other computers, we give in this appendix those features of the code that are specific for our system, and which perhaps have to be implemented in other ways for other machines.

### 19.1. The Timing Routine

It is essential that a timing routine is available. The call is of the type

CALL TIME (ICLOCK)

where the integer ICLOCK measures the computer time in units of 1/80 sec. The actual position of the zero on the time scale is irrelevant for our purpose.

### 19.2. The Random Number Generator

The random number calls are of the type

CALL RANDL (IX, R)

In our case, a standard multiplicative generator<sup>23)</sup> is used. IX is a current integer and is not used (yet it has to be initialized, cf. the input description in 6.1). R is the resulting random number between 0 and 1.

### 19.3. Input/Output Units

The symbolic nos. of the input and output units are 5 and 6 for the IBM 7094 at NEUCC. Only the statements

NIN = 5

NOUT = 6

(at the beginning of the code print-out, cf. Appendix III) should be changed to fit other conventions.

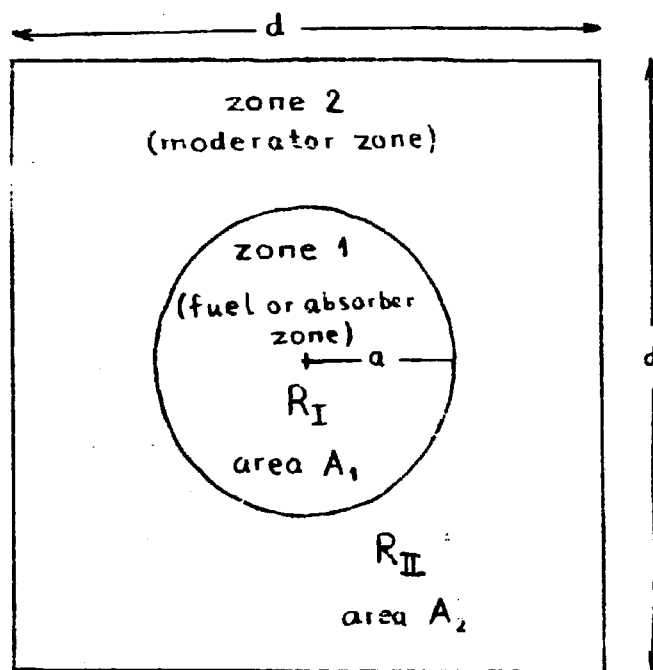


Fig. 1. MCSUP geometry

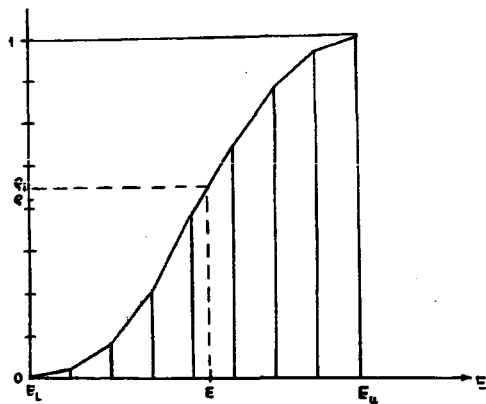
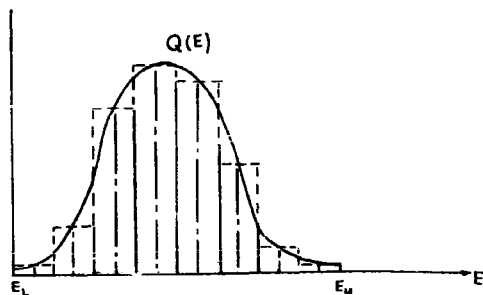


Fig. 2. Sampling of initial energy, schematic.

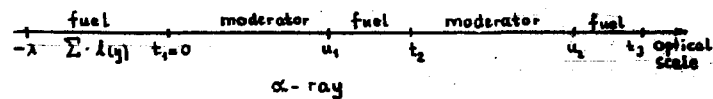
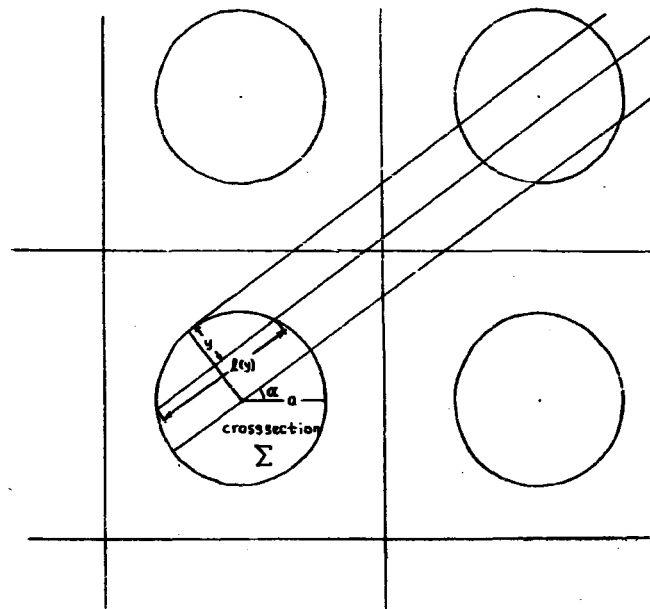


Fig. 3. Calculation of  $P^*$